DOCUMENT RESUME

ED 271 117 IR 051 554

AUTHOR Nipp, Deanna
TITLE In-House Library Production of Microcomputer Software for Freshman Orientation.
PUB DATE 12 Apr 85
NOTE 27p.; Paper presented at the Joint Conference of the CUNY Library Association and the Greater New York Metropolitan Chapter of the Association of College and Research Libraries (New York, NY, April 12, 1985).
PUB TYPE Guides - Non-Classroom Use (055) -- Reports - Descriptive (141) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS *Academic Libraries; *Computer Assisted Instruction; *Computer Software; Higher Education; *Library Instruction; Library Materials; Library Services; *Microcomputers; Program Descriptions
IDENTIFIERS Course Integrated Library Instruction; Mansfield University PA

ABSTRACT

This report describes a microcomputer tutorial that was designed, produced, evaluated, and revised by the library of Mansfield University of Pennsylvania and is now in use for new student library orientation. The report includes: (1) a description of the microcomputer program; (2) background on the production of the program and a discussion of the way it fits into Mansfield's total bibliographic instruction program; and (3) a description of the production process for those who are considering the use of microcomputers for this purpose. Reproductions of a series of screens demonstrate the content, organization, format, and interactive nature of the program. The design is based on educational goals. Scripting was done on graph paper to keep screens balanced, uncrowded, consistent in format, and easily communicated to the programmer. Even though library materials are being used as part of the package, maintenance has required minimal use of staff time. The tutorial is being assigned to students in freshman composition, developmental reading and writing, and introduction to education classes. (Author/THC)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy

ED271117

IN-HOUSE LIBRARY PRODUCTION OF MICROCOMPUTER SOFTWARE FOR FRESHMAN ORIENTATION

PRESENTED BY
DEANNA NIPP
COORDINATOR OF BIBLIOGRAPHIC INSTRUCTION
MANSFIELD UNIVERSITY OF PENNSYLVANIA

AT THE JOINT CONFERENCE
OF THE CUNY LIBRARY ASSOCIATION'S
LIBRARY INSTRUCTION COMMITTEE
AND
BIBLIOGRAPHIC INSTRUCTION
OF THE GREATER NEW YORK METROPOLITAN
CHAPTER OF THE ASSOCIATION OF
COLLEGE AND RESEARCH LIBRARIES

CUNY GRADUATE CENTER
NEW YORK, NEW YORK
APRIL 12, 1985

IR051554

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Deanna Nipp

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

PART I: DESCRIPTION OF THE PROGRAM

PART II: PRODUCTION BACKGROUND AND USE OF THE PROGRAM

- A. MANSFIELD UNIVERSITY LIBRARY
- B. ORIGIN OF THE PROGRAM
- C. DEVELOPMENT OF THE FIRST VERSION
- D. EVALUATION
- E. REVISION
- F. CURRENT USE

PART III: PRODUCTION PROCESS

- A. REQUIRED COMPETENCIES
- B. WRITING THE PROGRAM
- C. EVALUATION
- D. MAINTENANCE

Since January, 1981, the Mansfield University Library has designed, produced, evaluated and revised a microcomputer software package, Search and ... Find It!, for library orientation. Thank you for the invitation to share with you Mansfield's experience in automating a part of our instruction program. I began my library career by taking a degree at Pratt Institute and it is a real pleasure for me to be back among the librarians of New York.

This presentation will be in three main parts: first, a look at the microcomputer program itself; second, some background on the production of the program and the way it fits into Mansfield's total bibliographic instruction program; and, finally, a close look at the production process for those who are considering the use of microcomputers for this purpose and are wondering what is involved in putting such a program together.

1

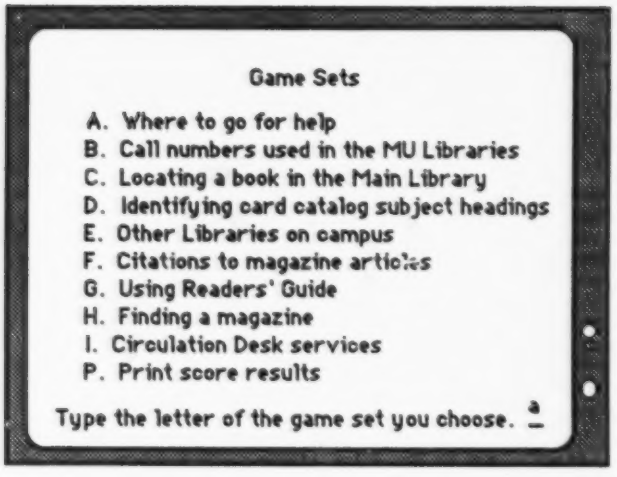
PART I: DESCRIPTION OF THE PROGRAM

First, we will look at the program. Physically, it is a disk packaged in a notebook which also includes information on loading the program and some sample pages from Readers' Guide to Periodical Literature. Twenty copies are available at the reserve desk, where students check them out and then take them to the University's main academic computing center, which is located in the library building.

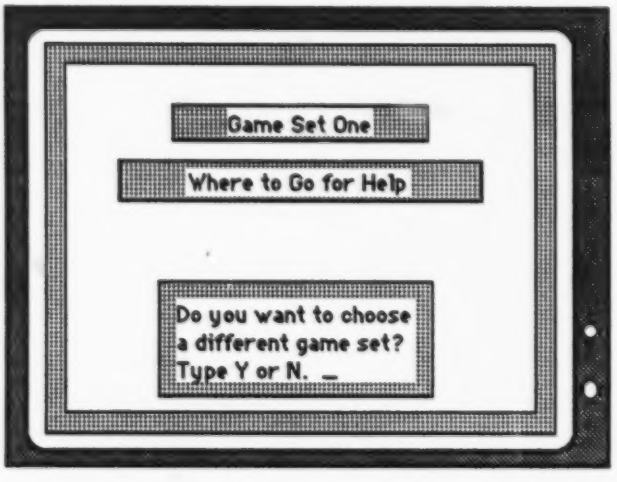
When students load the program, the title screen is the first screen they see:

The title screen and a credits screen are followed by several screens of introduction which are used to give a brief explanation of the content and organization of the program and to assure students that they need no special computer skills to make it run.

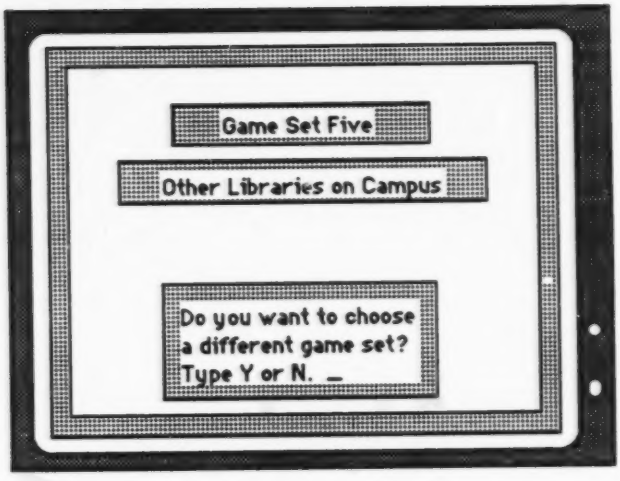
This introductory material is followed by a screen which lists all program sections. Students must finish all sections to complete the program, but may do them in any order. The "a" in the lower right hand corner of the following screen indicates the student running the program has selected the section titled "Where to go for help."


Game Sets
A. Where to go for help
B. Call numbers used in the MU Libraries
C. Locating a book in the Main Library
D. Identifying card catalog subject headings
E. Other Libraries on campus
F. Citations to magazine articles
G. Using Readers' Guide
H. Finding a magazine
I. Circulation Desk services
P. Print score results
Type the letter of the game set you choose. a

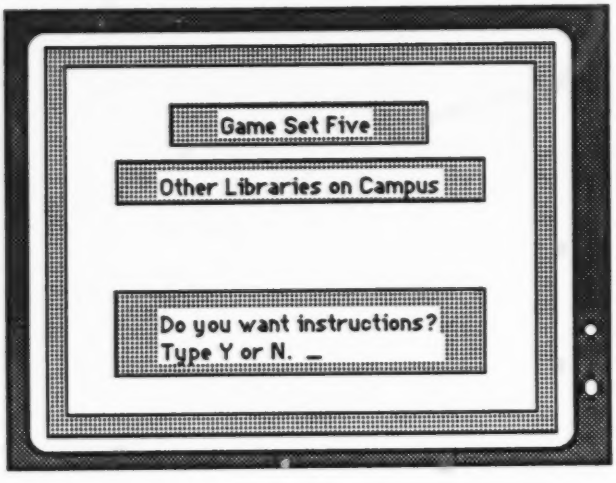
This leads to the title screen in the first section.



Typing "Y" in response to the question "Do you want to choose a different game set?" leads the student user back to the screen which lists all the sections of the program. A student could at that point select a different section, let us say, the fifth section.

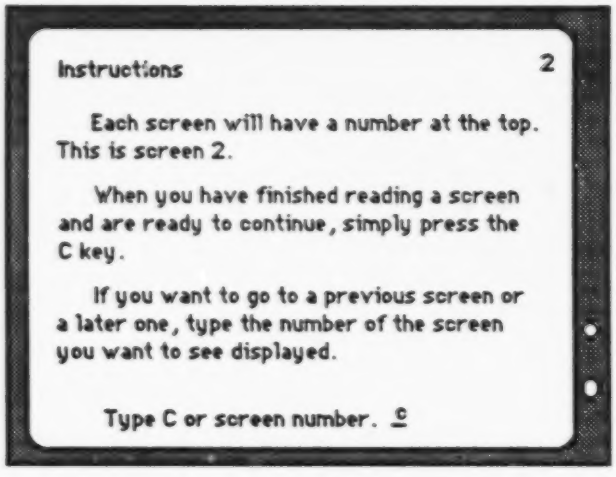


Typing "N" leads to the appearance of a new question on the screen.



Typing "Y" leads to the second screen in the section.

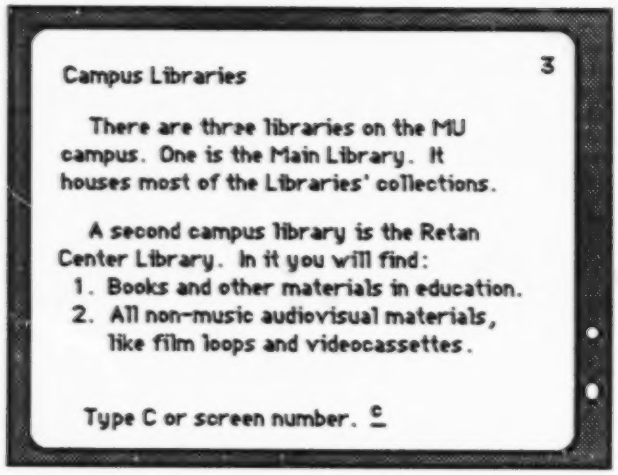
This is the second screen. Actually it is the same in every section. It is an important screen because it gives students an understanding of how they can control their learning experience.


Instructions 2
Each screen will have a number at the top. This is screen 2.
When you have finished reading a screen and are ready to continue, simply press the C key.
If you want to go to a previous screen or a later one, type the number of the screen you want to see displayed.
Type C or screen number. c

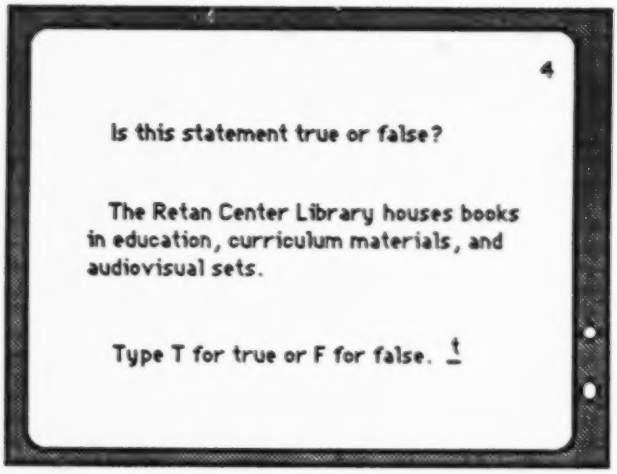
From information presented on this screen it becomes obvious that by always choosing the "C" key, one can continue in a natural progression through the lesson. But if a student feels he or she understands the material already or wants a review, it is easily possible to skip ahead or to return to earlier screens in the section by typing the appropriate screen number. In fact it is even possible to leave the section altogether by typing "1" (which would return the program to the title screen of the section) and then, in answer to the question "Do you want to choose a different game set?", typing "Y." Of course, this presents the screen with the list of sections, from which a new selection may be made. This arrangement gives the student a high degree of control over the learning experience.

Now let us continue into the lesson.

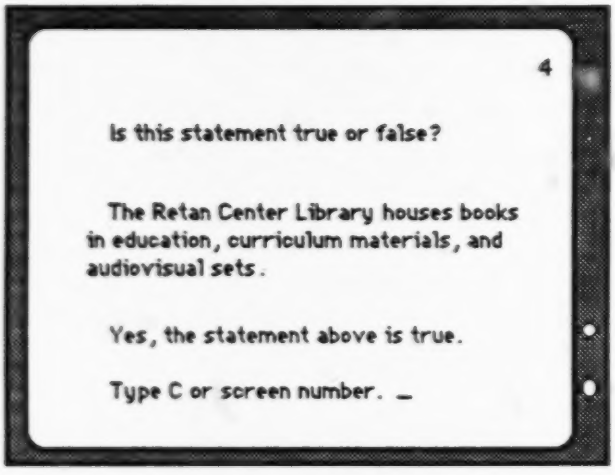
Screen 3 presents material to be learned.


Campus Libraries
3
There are three libraries on the MU campus. One is the Main Library. It houses most of the Libraries' collections.
A second campus library is the Retan Center Library. In it you will find:
1. Books and other materials in education.
2. All non-music audiovisual materials, like film loops and videocassettes.
Type C or screen number. c

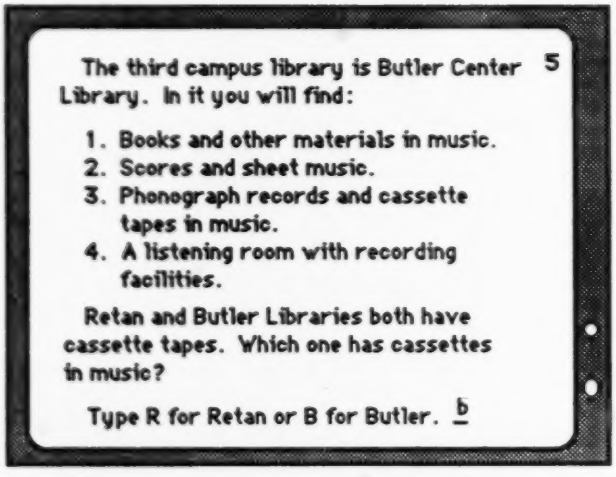
Screen 4 asks a question about it.


4
Is this statement true or false?
The Retan Center Library houses books in education, curriculum materials, and audiovisual sets.
Type T for true or F for false. t

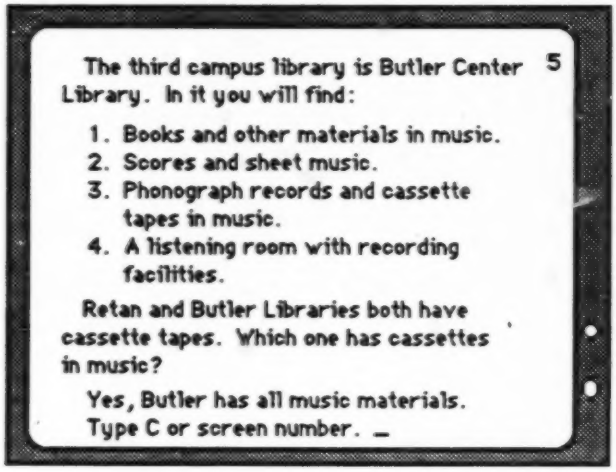
Since the answer typed in is correct, the program responds by removing the question line and replacing it with the information that the answer is correct and adding the instruction for continuing with the lesson.


4
Is this statement true or false?
The Retan Center Library houses books in education, curriculum materials, and audiovisual sets.
Yes, the statement above is true.
Type C or screen number. _

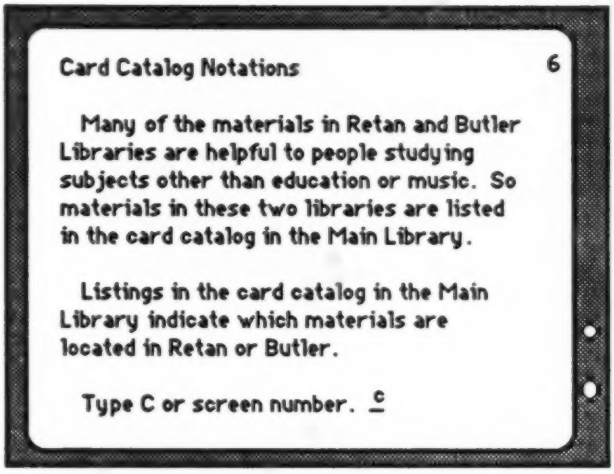
And the lesson continues.

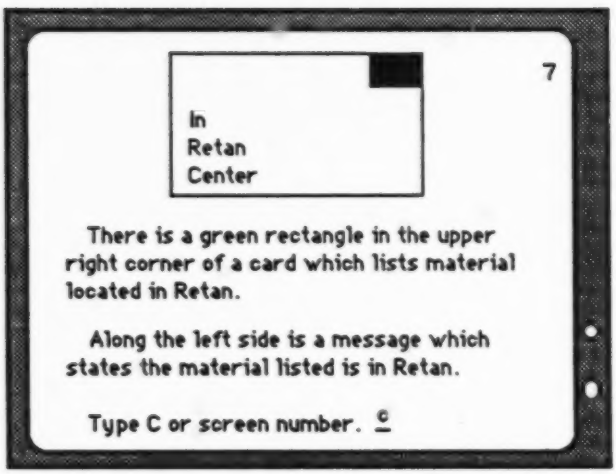

The third campus library is Butler Center Library. In it you will find: 5
1. Books and other materials in music.
2. Scores and sheet music.
3. Phonograph records and cassette tapes in music.
4. A listening room with recording facilities.
Retan and Butler Libraries both have cassette tapes. Which one has cassettes in music?
Type R for Retan or B for Butler. b

Again the response is correct and so the question is replaced with a note that the response is correct and the instruction for continuing the program.


The third campus library is Butler Center Library. In it you will find: 5
1. Books and other materials in music.
2. Scores and sheet music.
3. Phonograph records and cassette tapes in music.
4. A listening room with recording facilities.
Retan and Butler Libraries both have cassette tapes. Which one has cassettes in music?
Yes, Butler has all music materials.
Type C or screen number. _

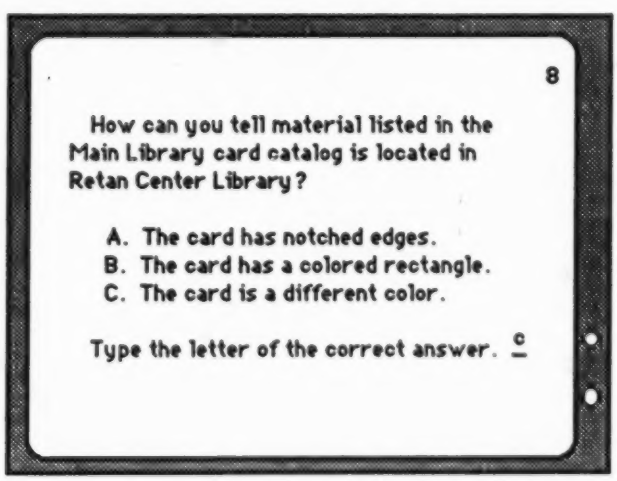
And instruction continues with the next screen.


Card Catalog Notations 6
Many of the materials in Retan and Butler Libraries are helpful to people studying subjects other than education or music. So materials in these two libraries are listed in the card catalog in the Main Library.
Listings in the card catalog in the Main Library indicate which materials are located in Retan or Butler.
Type C or screen number. C

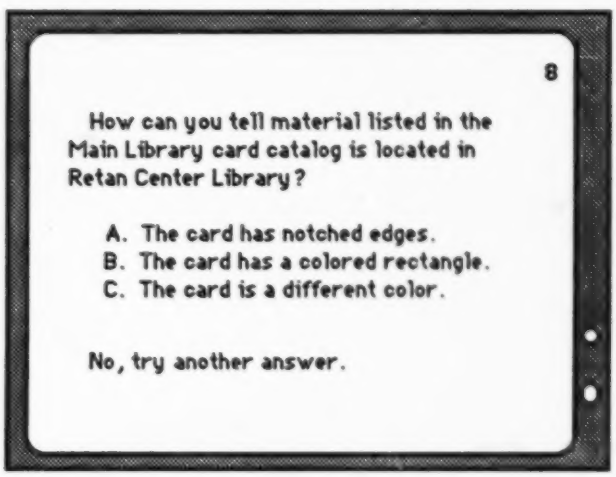

7
In
Retan
Center
There is a green rectangle in the upper right corner of a card which lists material located in Retan.
Along the left side is a message which states the material listed is in Retan.
Type C or screen number. c

This display, with green rectangle and green location message on the card catalog card, is replaced with one using blue on a card listing material in the Butler Center Library. The software we used for writing the program was chosen partly so that displays like this could be produced and enhanced with color.

The next screen asks for feedback.

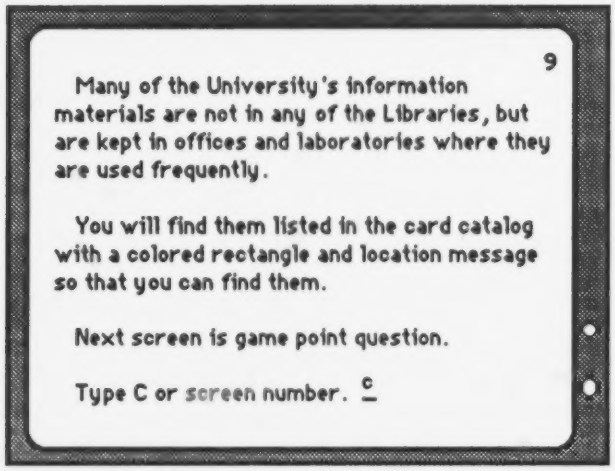

8
How can you tell material listed in the Main Library card catalog is located in Retan Center Library?
A. The card has notched edges.
B. The card has a colored rectangle.
C. The card is a different color.
Type the letter of the correct answer. c

This time a wrong answer is entered and the program responds accordingly.

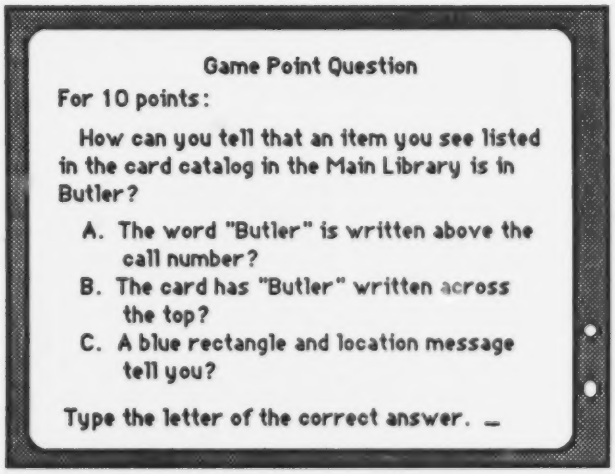

8
How can you tell material listed in the Main Library card catalog is located in Retan Center Library?
A. The card has notched edges.
B. The card has a colored rectangle.
C. The card is a different color.
No, try another answer.

The question line "Type the letter of the correct answer. _" reappears and the student must enter the correct response before the line "Type C or screen number. _" appears to allow continuation of the lesson. While students have a high degree of control over the learning experience, they do not have total control. For almost all of these interactive questions a student must enter a correct response before he or she can continue. As you can see, the design of the instruction is such that it is never difficult for a student to get the correct response. During several semesters of program use we have not had problems with students not being able to continue because they could not get the right answer.

Screen 9, which appears next, is the final screen of instruction in the section on campus libraries.


9
Many of the University's information materials are not in any of the Libraries, but are kept in offices and laboratories where they are used frequently.
You will find them listed in the card catalog with a colored rectangle and location message so that you can find them.
Next screen is game point question.
Type C or screen number. C

Each section ends in a question or series of questions to test student understanding of some part of the instruction. Students who give correct answers to these questions are given points which the microcomputer records for a printout which is produced at the end of the program. The message in the lower portion of screen 9--that the next screen is the game point question--warns the student user that this is the last screen of instruction in this unit and therefore the last chance to go back for review before getting the scoring question. If the student types "C," the next screen presents the "Game Point Question."


Game Point Question
For 10 points:
How can you tell that an item you see listed in the card catalog in the Main Library is in Butler?
A. The word "Butler" is written above the call number?
B. The card has "Butler" written across the top?
C. A blue rectangle and location message tell you?
Type the letter of the correct answer.

During the presentation of the game point question screens, control of the program sequence is again taken from the student. The student must enter an answer before continuation resumes. However, in the case of game point questions, which are designed to record student learning, students do not get repeated chances to give correct answers. If the wrong answer is typed in, a message appears that it is not correct and the student is asked if he or she would like to repeat the section. Of course the second time through the student may skip over much of the material and go directly to the last instructional screen and then to the game point question or, alternatively, read the section more carefully if that is deemed necessary.

If student response to the game point question is correct, a message appears on the screen to notify the student that the answer is correct and then the game point question screen is replaced with a screen on which a clown appears. The clown holds a sign with a message such as "Slick work!" or "You called it!" These clowns were included to give a graphic reward for right answers and therefore increase motivation to learn. However, they take so long to build in line by line sequence on the screen that they slow student progress through the program and have become negative rather than positive motivators. When the program is revised, the clowns will be replaced by something which takes less time to appear, is more varied in image, or perhaps moves in some way.

Most sections end with a game point question which can be answered as the student sits at the microcomputer. In two sections, however, the student is assigned an actual search for material in the Library; one search is for a book and one is for a magazine article, which the student must identify using Readers' Guide to Periodical Literature and find using the Libraries' serial location list. Assigning actual searches adds significantly to the time needed for program production and also requires that the materials used be monitored regularly to assure their continued presence in the correct location. This has produced no significant problems for us. A student assistant assigned to the Reference Department is given a list of all these materials and checks and cleans the writing from them regularly throughout the year and more frequently during periods of heavy use. (Pages assigned from Readers' Guide and magazines were liminated in preparation for their heavy use.)

When a student has finished the program, a printout of scores is produced in duplicate. Students can use one copy to hand in to the instructor who has assigned Search and ... Find It! and turn in the second copy to the Reference Desk. We use these to monitor student success, evaluate the program, collect statistics, and maintain a file so that we can give a copy to any student who is given Search and ... Find It! as an assignment in another class.

Now, having taken a look at the program itself, let us turn to the Mansfield University Library for background on the production process.

PART II: PRODUCTION BACKGROUND AND USE OF THE PROGRAM

Mansfield University Library

Mansfield University is a state-owned institution of higher education in Pennsylvania. It has an enrollment of about 2800 students, with 800 to 900 new students entering each fall. It offers a variety of liberal arts and professional degrees, mostly at the bachelor's level with a few at the associate and master's levels.

There is a library staff of eight librarians, seven clerical workers, and the director. Staff numbers have been decreasing during the last ten years due to shrinking budgets, but we have taken advantage of automation in an effort to maintain or improve service. The quality of the staff is excellent; I feel fortunate to be able to work with this group. It holds very high standards of service and is progressive in its outlook for providing service to our user group.

Mansfield's bibliographic instruction program is three-tiered, the first level being orientation to the organization and services of the campus libraries, the second being an introduction to reference sources including the Wilson indexes, and the third being an in-depth look at the research tools and strategies in a student's major field. (I say it is three-tiered and that is our goal, but in fact we have not been totally successful at delivering it across the entire curriculum.) In January, 1981, orientation level instruction was being presented in a non-credit university orientation course required of all entering students. There were usually about twenty-eight sections, most of which were met by the four public service librarians who worked in the Main Library. All but four sections met during reference desk lunch hour times and we met three to four sections at a time. Instruction was in the form of a short tour, a lecture, and finally an exercise to give students an active learning experience. The exercise included two actual searches, one for an assigned book and one for an article on an assigned topic from a specific volume of Readers' Guide to Periodical Literature.

Origin of the Program

The idea to try computer-assisted-instruction (CAI) came in January, 1981. No other unit on campus was making plans to provide microcomputers. The librarians agreed they should be made available and made an initial commitment to work for funding for two Apples. Out of these discussions came the idea to try using the microcomputers for library orientation.

Computer-assisted instruction offered a number of advantages, some generally applicable and some specific to our own situation.

1. The learning experience can be tailored to each individual student's needs.
2. The rate of progress can be self-paced.
3. There is immediate feedback to student answers.
4. Computers have endless patience and no prejudices.
5. Offering instruction in this format could ease some of our scheduling problems.
6. Staff time could be reserved for more advanced instruction, which must be tailored for each class.
7. Student attitude concerning the university orientation course was poor. It was hoped using a medium students enjoyed would help create conditions more favorable for learning.

There were also some very obvious disadvantages to adopting computer-assisted-instruction.

1. Students would not meet a reference librarian. It is the experience of us all that many students never approach us at the reference desk until they have had us in class.
2. One major goal of our orientation-level instruction is an introduction to the organization and services specific to the Mansfield University Libraries. There was obviously no commercial or public domain software available which would do this.

Since I was eligible for a one-semester sabbatical leave and was willing to use it to learn BASIC and produce the program, the Library Department felt it would be a good use of resources to try to develop a CAI module for orientation, use it, and evaluate it. In the spring semester of 1981 I wrote the sabbatical proposal and our director worked on funding for two microcomputers.

Development of the First Version

In July, 1981, the University supported the Library's proposals by granting the sabbatical leave for the fall semester of 1982, supporting funding requests for equipment, and later giving a special grant for software and supplies for the orientation tutorial. In December, 1981, our first two microcomputers arrived and in January, 1982, I took a week-long workshop in BASIC.

In the fall of 1982 work on the program began in earnest. Six weeks were devoted to learning BASIC through a self-instructional workbook. To learn about educational design using computers, I read material available in our education library and visited Pat Arnott, who had designed a library orientation program at the University of Delaware using PLATO and their mainframe equipment. It took nine weeks to design, script, and program the software and write an evaluation instrument. It took another three weeks to develop "documentation," the accompanying printed information which included instructions for loading and a sample copy of our serials location list, and to prepare the library materials for assigned searches.

While I was the one with ultimate responsibility for bringing the project to fruition, support came from many other people, all of whom were important in its successful completion. Our academic computer services department was not staffed to the level that it could teach us design and program the unit; but it offered invaluable assistance in the selection of software and the recommendation of a student who helped in the final stages of programming. Students and faculty were supportive in my effort to learn BASIC. A professor in the Special Education Department acted as consultant on evaluation of the project. A library student assistant produced drawings for our loading instructions. I learned that while it is possible for a library itself to produce orientation software, there are many different skills involved and it is important to find ways to gather the variety of personal resources needed to produce the final product.

The result of the sabbatical semester was Search and ... Find It!: A Game of Orientation to the Mansfield State College Libraries. It was programmed in BASIC for use with the Apple, covered all the material you saw on the screen which listed program sections, required two actual searches for library materials, and produced a printout of student scores. In short, I felt it was "marvelous." And marvelous it was, too, if you consider that it was a major project and a first attempt. It was also "home-movie" quality: it was much too wordy and did not take full advantage of the interactive nature of the equipment we were using. These are common problems for beginning software writers.

Evaluation

The program was first used in a pilot study in the spring semester of 1983. Each college orientation section was divided, half of the class receiving their instruction in the lecture-exercise format and half receiving microcomputer CAI. Evaluation was done by administering a formal pre- and post-test to both groups, asking for student reaction, and observing use. Results showed the program met our educational goals. Microcomputer students' post-test scores improved to the same extent as control group lecture-exercise scores. Comments from our students showed they truly enjoyed it. Observation of use convinced us students were able to understand instructions as they were written and that there were no "bugs" in the programming. Only one major problem surfaced. It took most students two hours to complete what we intended to be an hour long learning experience.

Revision

Just as I was completing the first version and even before it was evaluated, the institution made a commitment to acquire a large number of microcomputers and place them in an area of the Main Library, establishing it as the academic computing center. This was good news, for it would give us all the microcomputer time needed to reach all entering students. But the units selected were IBM PC's.

Because of the length and relatively low interactive nature of the first version, I recommended that instead of having the Apple version translated for the IBM PC we completely revise it for the new equipment. This work was done in the fall of 1983. I was released from some of my assigned departmental responsibilities in order to have the time to complete the new version. I had already thought out an improved design; the scripting took 100 hours of time; preparation of documentation another 25 hours. We hired a computer and information science student at $3.35/hour for 100 hours to do the programming. PILOT language software, which cost $400 and will get campus use beyond ours, was used for the programming. Supplies for twenty copies of the package cost $200. These materials included notebooks, disks, paper, and laminating film.

The revision resulted in a program of much improved quality. It takes most of our students only an hour to complete it. It is much more interactive. More color and graphic displays are used. It is essentially a professional production, one which we can provide for our students with pride.

Current Use

During the final weeks of the revision process, the institution made the decision to discontinue the orientation course. Consequently we were in the position of having a good vehicle for delivering orientation level bibliographic instruction with a minimum commitment of staff time, but no point in the curriculum for delivering it to all entering students.

We have worked on implementing its use in the same ways college and university librarians promote library orientation in any other format: (1) talking with departments which offer beginning level courses which require library research; (2) recommending to curriculum committees that it be incorporated into the institution's general education program; and (3) enlisting the support of the administration. In the process we have learned that one advantage in offering course-integrated library instruction in microcomputer format is that many classroom faculty members are attracted to the idea of being able to provide for library instruction without having to commit class time.

Statistics based on the number of printouts turned in at the reference desk and a survey of faculty members who used the program as an assignment in their classes showed that 400 students used Search and ... Find It! in the 1984-85 academic year. This is close to the number we were reaching with the lecture-exercise experience in the college orientation classes. Printouts showed that users were from the Composition I and Speech I classes and from the developmental reading and developmental writing classes. A significant number of students used it on their own without having it assigned in any class.

In summary, our experience shows that even within the context of a growing, changing institution, it is feasible for an undergraduate library to develop the personal expertise and gather the material resources necessary to develop a microcomputer tutorial for library orientation. And it can be successful. However, because of the time-consuming nature of the project, it is important to first look for available software which meets the stated educational goals and finally commit to the production of a microcomputer tutorial only if expected use justifies the staff time needed to produce the package.

PART III: PRODUCTION PROCESS

Required Competencies

Production of a microcomputer tutorial requires competencies in three separate areas: content, educational design, and programming.

Content offers no problems when the software to be written is in the field of one's expertise. Librarians know and teach research skills daily; what is to be taught in bibliographic instruction is material we already have mastery of. In fact, we are the best ones to organize this information for students at our own colleges and universities, for through experience with them we are aware of the areas that cause the most problems and what descriptive terms and phrases work best in helping them find their way around our libraries.

Educational design, which has to do with using screen space, computer memory, computer logic, and the interactive nature of the equipment to teach, offers a new challenge. There are three ways to develop this competency. The first is to take a course in educational design, but particularly one with a sizeable segment on CAI. The second is independent study through reading. Recently three good books on designing CAI have been published. I have found Robert L. Burke's CAI Sourcebook (Englewood Cliffs, N.J.: Prentice Hall, 1982) to be especially useful. Furthermore there has been a good deal of research done on effective use of computers for teaching. It addresses questions such as the most effective sentence structure and length, the use of color, and the use of graphic rewards. Many of these reports can be found through a search of Current Index to Journals in Education and a look at the monographic holdings of an academic library serving the field of education. Finally there is the review of other software. This is of paramount importance for through this process one begins to understand the interactive possibilites available, sees good examples of the use of this medium, and also discovers firsthand those things which don't work well or lead to frustration. It is important in the reviewing of software that not to go through a program putting in only correct answers; entering incorrect and even non-sensical answers tests software and demonstrates how different programs handle these events.

Finally one must provide for programming competencies. Basically there are two choices: learn a programming language ourselves or identify a person outside the library who will do the programming. Learning a language may be an altogether new experience and at times seem like a struggle. But it is not beyond our abilities and becomes fun early on as we begin to manipulate what appears on the screen and to see applications for an instructional unit. Having learned independently, I would recommend taking a course so that a resource person is available for working through problems. My learning experience was successful partly because people on campus and a family member (my son) were very willing to answer questions as they came up.

The alternative is finding a programmer outside the library organization. The institution's academic computer services unit may be staffed to provide this service or there may be a computer science major offered. Many computer and information science students are eager to use their newly developed skills on a real life project and to earn a little money. (If funds are a problem, one alternative is to explore whether such a project would qualify for indepent study or similar academic experience.) Mansfield used a student for the second version. The arrangement worked out well, because it saved my time for design and scripting The student, Kevin Roberts, took one-hundred twenty pages of script home over winter break and three weeks later presented us with a piece of software which met our specifications and ran without hitch.

It is important in selecting a programmer to find someone who can communicate well and is willing to make the designer of the program aware of the full range of possibilities and limitations of the system being used. Kevin and I worked together closely in the initial stages of scripting and he was able to point out features of the equipment I might want to use and didn't seem to be aware of and also to let me know where I had set up some input devise which would take hours to program but could easily be replaced with something simpler. This early communication was the basis for his being able to take the script and with very few questions complete the programming independently.

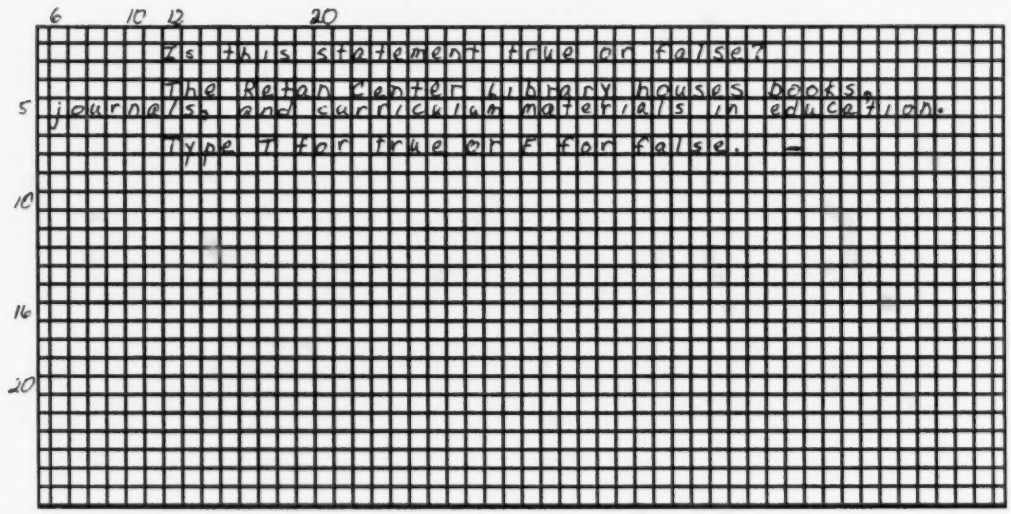
Writing the Program

The important first step in the writing process is making a list of educational goals. This list will be the basis for selecting the design (organization) of the program, will serve to focus all writing, and will become the basis for later evaluation.

The second step is to identify the most appropriate functional design. Some standard types are (1) drill and practice, which lends itself to a unit like one which teaches multiplication tables in which basic concepts have already been presented and the unit provides for the repitition which supports memorization; (2) simulation, which is presently being employed as a substitue for costly laboratory experiences; (3) problem-solving, which lends itself to courses such as those in statistics; and (4) the tutorial, which works best for initial presentation of new material and is used with highly verbal material.

One must then decide on the physical design. Some options are (1) linear, in which all students take the same route through the program from start to finish; (2) spiral, which flows in a spiral through the material to be learned, perhaps explaining a process and then applying it in a series of examples; and (3) multi-track, which is written on several levels to accommodate the differing abilities of student users. Mansfield's program is a modified linear tutorial. Instruction is presented in units, all of which a student must complete; and though there is an obvious order of presentation students do not have to stay with it. There is one area where students are given the opportunity to test out of instruction; and the section which covers the serials location list gives some drill and practice.

With goals written and an overall organization or design of the program thought out, the scripting process can begin. The script for the program is written screen by screen. And the easiest way to do it is with a pencil and eraser on graph paper which has the same number of spaces across and lines down as the invisible grid on the screen with which one is working. In this way it is possible to communicate to the programmer--even if it is yourself--what should be on each screen and precisely where it is to appear. The process is further facilitated by using graph paper which is wide enough to provide space at the side to write special instructions about what the program is to do next on the basis of any given student input. Doing the work with a pencil makes it easy to change what is written, in the event there is not enough screen space to develop a planned concept or the use of available screen space must be modified to accomodate an idea to be included there. The following display is a portion of one of our scripting sheets.



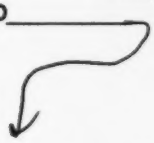
Each sheet should be labelled with identifying marks which indicate where it is located in the program, so that as the programmer works, moving from sheet to sheet and sometimes back, it is always quickly possible to regroup in sequence. Our sheets were labelled at the top with section name and screen number.

The "_" at the end of the display above indicates to the programmer (and eventually the user) that input is expected. When user input is typed in, the program has to have instructions concerning what to do with it. And they must cover every possibility, for typing errors and deliberate attempts to foil the system happen. These are the instructions which accompanied this scripting sheet. The numbers in parentheses tell the programmer where responses are to appear.

If input is anything but T or F, erase immediately and get input again.

If input is F type (at 12,9)
Look again. The statement above is true.
Pause. Erase "Type T for...etc." line and "Look again..." line.
Pause. Replace "Type T for true..." line.
Get input.
Go to_____

If input is T type (at 12,9)
Yes, the statement above is true.
Go to



Type (at 6,16)
To continue press C or screen number. _

If input is not C or screen number, erase it and get again.
If C, go to screen 5.
Otherwise go to screen specified by number.

Scripting is great fun. Here are some of the guidelines I set for myself as I worked.

1. Keep the screen visually balanced.
2. Do not crowd. Present only one or two concepts per screen or one concept and one question.
3. Leave room at the bottom free for instructions, i.e. "Type C or screen number. _"
4. Set up a consistent format so the user is not constantly surprised by the presentation and is free to concentrate on content. Search and ... Find It! uses the same format for the title screen of each section, a second format for instructional screens, and a third for game point question screens. It remains consistent in its use of the line "Type C or screen number. _" for continuing the lesson.
5. Keep the educational goal being written for uppermost in the mind and never write one word not absolutely necessary for student understanding of that goal. Keep the sentences short. This is of paramount importance in a tutorial, where every word adds to the time necessary to complete the program and where every unnecessary sentence detracts student attention away from the concepts important in reaching the goal.
6. Make keyboard manipulation easy. A student who has not used a keyboard will not be able to concentrate on content if he or she is having to spend a lot of time searching for what key to press. For the most part this can be attained by asking for

input of only one or two letters and limiting the use of any keys which are not the letter or number keys.

7. Remain consistent in the choice of any necessary technical terms. The first time I introduced our serials location list I called it the "red book" because that is what students will hear it called most often in the library. From that point in the program it is always called the "red book," never the "serials holdings list" or the "serials location list."
8. One of the most powerful advantages of CAI is the opportunity for immediate feedback for student response. Eventually I worked out four parameters for response to a student answer.

    a. It must clearly state if the answer is right or wrong.
    b. It must be done in the fewest number of words possible.
    c. If the student answer is incorrect, the response must be encouraging and supportive--that is, I would not script "You're wrong.", but instead "The answer is wrong." And often I would add something like "Let's try again."
    d. If the student answer is incorrect there must be a mechanism for the student to discover the correct information.

9. Each time input is requested, specify to the programmer what the program should do to respond to right answers, wrong answers and non-sensical input.

With scripting done, programming can begin. It can be done directly from the scripting sheets. Using BASIC language, the programming itself is done line by line and I found it quite easy to program right from the script generally using a program line for each line on the screen and adding extra program lines to handle input.

The final step in writing the program is producing documentation. Documentation is material on paper which will explain the program to a user and to anyone with subsequent responsibility for revising it. It includes the scripting sheets and the computer coding produced by the programmer. It should also include a description of the program for display in the software package. The description includes the title, credits, date of production, content, and exact statement of the equipment needed to run the program.

Other documentation will depend on the nature of the program. For Search and ... Find It!, instructions for loading are included in the software package because we have a significant number of entering students who have not yet used a microcomputer and we want them to be able to use the program independent of our help. Copies of several pages from Readers' Guide to Periodical Literature are also included because many of our entering students have never seen it before and we feel it is important for them to see the format. Two of the interactive questions require the use of these pages.

Since there are two actual library searches assigned in the program, the materials provided for these searches are also a part of the documentation. These include five special cards in the card catalog and five book blocks which have been assigned call numbers and are shelved in our stacks. They also include laminated pages in our shelf set of Readers' Guide and laminated pages of magazine articles in some of the bound periodicals in our stacks.

Evaluation

Evaluation is an important part of any instruction, but even more so with computer-assisted-instruction, for students will be attempting to learn from our software while instructors are not present to monitor the experience. While we may never be able to prevent every problem, we must be quite sure that we have done everything possible to identify and eliminate those aspects of the program which are generally going to cause difficulty. There are several ways to evaluate and we can expect to learn different things from each method.

First is a formal evaluation to test for student mastery of the educational goals of the program. One way to do this is through a pre-test and post-test using a control group which has had other instruction and even one which has had no instruction. We found that consulting with a faculty member who had expertise in this area was invaluable, for he not only helped write meaningful questions, but was also able to do a statistical analysis of the results. Our only problem was getting access to class time for administering the tests.

A second method is to ask for student comment. This can be done as part of the program itself or by distributing questionnaires. Both versions of Search and ... Find It! end with a request for comments on the program and it was through this means that we first realized the first version was much too long. Not all comments are negative criticism, however, and the positive ones can be quite encouraging. Almost without exception we found comment to be open, honest, and useful. We also ask for students' opinions when they turn their printouts in at the Reference Desk. Their answers are frank and consistent: "I liked the chance to use the micro.," "Yes, I really do understand more about how to find information in this library," and "Get those clowns out of there."

A third method is to observe use. This reveals those points where students are having a difficult time understanding the instruction as it is written or understanding what they are supposed to do next to make the program go. They may not remember these points after they finish and so can't really tell us at the Reference Desk. Mansfield is fortunate to have the microcomputers we use located in our building. We quickly become aware and can respond to any problems. The equipment which hooks four of the microcomputers to one printer became defective last year and since we learned about it very early we were in a position to write ancillary instruction sheets for loading and thus prevent students from working through the entire program and then lose their scores.

Maintenance

While use of a microcomputer program can free a good deal staff time, there is still a need to set aside some time for maintenance. The first consideration is the maintenance of the disks. The floppy disks used with microcomputers are relatively sturdy and stand up to a lot of use. But there are times when they are bent or get lost and eventually they do wear out So it is imperative that one keep back-up disks. This is almost the first rule one learns when starting to write programs and it is even more imperative when working disks are to be circulated for general use. During the first semester that the revised version of our program was in use, a clown on one of our disks was mysteriously replaced with the drawing of a nuclear explosion and the message "I got nuked." I shall be eternally grateful that we had back-up disks and I continue to keep two masters in a safe place where only the programmer and I can get to them.

I also learned it is convenient to have a number of extra copies. When a student reports that there is a problem with a disk, I can quickly have a cataloging label attached to one of the extras and keep the "problem" disks for review at a later time.

We have found no need to regularly check the notebooks the disks and instructions come in, but because there are a number of library materials used we have made provision to have them checked regularly to see that they are on the shelf where they belong and to remove any writing which has appeared. This is done with very minimal use of my time; I give a student assistant a copy of the materials involved and assign him or her the responsibility to check the materials more or less frequently throughout the semester depending on the level of use at any given time. During the 1984-85 academic year one book block disappeared on a regular basis; our student assistant had a natural talent for finding it.

The only other use of librarian time for maintenance of the system is keeping a file of the printouts. We make copies for any student who has completed the program once and then has had it assigned in a second course. We also use them to monitor use and collect statistics. This is much less time consuming than grading the exercise sheets we were using previously.

IN CONCLUSION

I do not think of the microcomputer as the ultimate tool for delivering freshman library orientation, but I do believe that it can be useful. My experiences lead me to believe that most academic libraries in which there is an interest can produce a program of reasonable quality, that students will learn from it and enjoy it, and that after the initial major commitment for production, it can be run with minimal need for staff time.

Thank you for this opportunity to share what has been for us at Mansfield a truly exciting and useful project.